数值分析第五次上机练习实验报告

——数值积分与数值微分

力 4　杨昊光　2014011619

一、问题描述

试用不同数值积分方法计算$I(f)=\int_1^3 f(x)\mathrm{d}x$的近似值，其中$f(x)=\frac{1}{x^2}\sin\frac{2\pi}{x}$.

参考值：$I(f)=-0.238732414637843\ldots$

1. 把[1,3]分成 4 个子区间，用五点 Gauss-Legendre 求积公式的复合求积公式计算。
2. 用 Romberg 求积算法计算积分，取$\varepsilon=10^{-7}$，并与第一种办法比较。

二、方法描述

1. Gauss-Legendre 求积公式与复合 Gauss-Legendre 求积公式

设$[-1,1]$内$(n+1)$阶 Legendre 多项式的零点为$t_0,t_1,t_2,\ldots$，即

$$P_{n+1}(t_k)=\frac{1}{2^{n+1}(n+1)!}\frac{\mathrm{d}^{n+1}}{\mathrm{d}x^{n+1}}[(x^2-1)^{n+1}]\bigg|_{x=t_k}=0\ \ ,k=0,1,2,\ldots,n.$$

现欲将$\int_a^b f(x)\mathrm{d}x$表示成$\sum_{j=0}^n A_j f(x_j)$的形式，且代数精度达到 $2n+1$，利用 $P_n(x)$的正交性，有：

$$A_j=-\frac{a_{n+2}}{a_{n+1}}\frac{(P_{n+1},P_{n+1})}{P_{n+2}(t_j)P'_{n+1}(t_j)}=\frac{2}{n+1}\frac{1}{P_n(t_j)P'_{n+1}(t_j)}.$$

对于一般区间$[a,b]$，可作 t_j 到 x_j 的变换：$x_j=\frac{a+b}{2}+\frac{b-a}{2}t_j\in[a,b]$，得到$[a,b]$上的求积节点。

复化该方法，将$[a, b]$等分成 m 个小区间，并在每个小区间再取求积节点并应用该公式，得到复合 Gauss-Legendre 求积法，可以提高精度。在本例中，取 $m=n=4$，有：

$$P_4(x)=\frac{35x^4}{8}-\frac{15x^2}{4}+\frac{3}{8}\ \ ,P_5(x)=\frac{63x^5}{8}-\frac{35x^3}{4}+\frac{15x}{8};$$

$$x_{jm}=\frac{t_j+3+2m}{4}\text{为第}m\text{个区间上的求积节点，}m=1,2,3,4;$$

$$\int_1^3 f(x)\mathrm{d}x\approx\sum_{m=1}^4\sum_{j=0}^4\frac{2}{5}\frac{1}{P_4(t_j)P'_5(t_j)}f(x_{jm}).$$

2. Romberg 求积算法

该方法包含两部分：重复利用梯形公式并减半加密网格，和利用 Richardson 外推法的思想进行加速。

减半加密网格和梯形公式：记$h_j=2^{-j}(b-a)$

$$T_1^{(0)}=T_{2^0}=\frac{b-a}{2}[f(a)+f(b)],$$

$$T_1^{(k)} = T_{2^k} = \frac{1}{2}\left[T_1^{(k-1)} + h_{k-1}H_{k-1}\right]，其中 H_j = \sum_{l=1}^{2^j} f\left(a + \left(l - \frac{1}{2}\right)h_j\right).$$

再对 T_1 进行 Richardson 加速：

$$T_{j+1}^{(k-1)} = \frac{4^j T_j^{(k)} - T_j^{(k-1)}}{4^j - 1} \quad ,j,k = 1,2,\ldots$$

若$\left|T_j^{(0)} - T_{j+1}^{(0)}\right| < \varepsilon$，则计算已达到精度，终止加密迭代，输出$T_{j+1}^{(0)}$为积分的近似值。

三、方案设计

采用 Matlab 符号计算功能简化程序，首先定义 f 为所求函数备用，用迭代法生成直到 5 阶 Legendre 多项式备用。

将[1,3]等分为 4 个小区间，作 $P_5(x)=0$ 的根排序后到 j=1:4 四个小区间的变换结果，存为 x(:,1:4)，将未经变换的值存为 x(:,5)，同时计算出系数 A，由于 A 不随区间变化，可以重复利用。同时求出每个积分节点上对应的函数值，存为 fk(n,j)，并将每个小区间上的积分值算出为 I1(j)=dot(A,fk(:,j))，最后将每个小区间的 I1(j)加起来得到[1,3]上积分的估计值 IGL。

Romberg 算法在实际操作时应该注意顺序。定义 eps 为结果容差并给出初始 T(1，1)的值，fh(l,k)为对应 k 的积分节点上的函数值构成的数组，按列求和得到相应的 H_k。对每次 k 加 1，刷新 fh 数组，求得 T(k,1)，再逐次利用 Richardson 加速公式推出 T(1,k)并与 T(1,k-1)作差比较误差大小，若误差小于 eps 则终止迭代，IR=T(1,k)为所求积分的估计值。由于 fh 的数据量随 k 的值增大呈指数增长，因此当 k≥10 时运算已相当缓慢，故只有当需要时才增加 k 的值并计算 T(1,k)。

最后是用内部符号运算的积分函数产生参考值并作误差分析。

四、计算结果及其分析

参考值为 $I(f)$= -0.238732414637843…

通过复合五点 Gauss-Legendre 求积算法得到的结果为

$$\int_1^3 \frac{1}{x^2}\sin\frac{2\pi}{x}\mathrm{d}x \approx -0.238732340343646 \ ,$$

绝对误差为$7.429419685012206 \times 10^{-8}$，相对误差为$-3.112028040382632 \times 10^{-7}$.

通过 Romberg 求积算法，取$\varepsilon = 10^{-7}$，循环 7 次，得到的结果为

$$\int_1^3 \frac{1}{x^2}\sin\frac{2\pi}{x}\mathrm{d}x \approx -0.238732414621624 \ ,$$

绝对误差为$1.621944245577822 \times 10^{-11}$，相对误差为$-6.793984168586034 \times 10^{-11}$.

可见 Romberg 算法得到的结果精度显著优于 Gauss-Legendre 复合求积公式的结果。然而 Romberg 算法整个过程取了 257 个[1,3]之间的点的值，而复合五点 Gauss-Legendre 算法只取了 17 个积分节点进行计算，因此复合 Gauss-Legendre 算法在对于已知数据较少的情况或对总计算量和时间要求较为严格的情况下使用会有相对比较好的效果。

五、结论

常用的数值积分方法有插值法，即求已知数据点的插值函数在对应区间上的积分，有梯形公式和 Simpson 公式等形式，在等距节点高次情况下有数值不稳定性，因而通常采用复化插值求积的方法，即把区间等分为若干个小区间，然后再每一个区间上求积分再求和，这样可以消除 Runge 现象的影响。对于 Gauss-Legendre 复合求积公式，其权函数为 1，而且利用 Legendre 多项式的零点作为积分节点的选取依据，最大程度上简

化了计算并优化了精度，有较大的应用潜力。

对于收敛慢的复合求积公式，利用 Richardson 外推的加速方法可以使之收敛更快，这也就是 Romberg 求积算法的依据。Romberg 算法通过减半加密网格和 Richardson 加速，可以将原本收敛很慢的复合梯形公式变得具有实用价值，在经过较少循环次数后就可以达到较高精度。

Romberg 积分算法对数据点的密度要求较高，运算量也较大，每一次循环精度提高也更快；相比之下复合 Gauss-Legendre 积分算法只需要特定位置的相对较少的积分节点，计算量也较小，是一种比较稳定的数值积分方法。可以说这两种方法各有所长，需要看应用背景具体选取。